

CS 665 Information Delivery on the Internet Final Exam - Name: _____
Fall 2003

Part 1: (75 points - 3 points for each problem)

- (A) 1. What protocol is used by all Web communications transactions?
(A) HTTP (B) DHCP (C) FTP (D) PPP
- (D) 2. Which is a Web browser?
(A) Apache (B) Tomcat (C) IIS (D) lynx
- (A) 3. How a link can be open in a new browser window in HTML?
C (A) `` (B) ``
D (C) `` (D) ``
- (B) 4. What is the correct HTML for making a checkbox?
(A) `<input type="check">` (B) `<input type="checkbox">` (C) `<check>` (D) `<checkbox>`
- (C) 5. What is the correct HTML for making a text area?
(A) `<input type="textarea">` (B) `<input type="textbox">`
(C) `<textarea>` (D) none of above
- (B) 6. What does CSS stand for?
(A) Casting Style Sheets (B) Cascading Style Sheets
(C) Creative Style Sheets (D) Colorful Style Sheets
- (D) 7. What is the correct HTML for referring to an external style sheet?
(A) `<style src="mystyle.css">` (B) `<style type="text/css" src="mystyle.css">`
(C) `<stylesheet>mystyle.css</stylesheet>`
(D) `<link rel="stylesheet" type="text/css" href="mystyle.css">`
- (D) 8. Which CSS tag is designed for a phrase in a line to appear in a different font size or color?
(A) `<section>` (B) `<div>` (C) `<division>` (D) ``
- (A) 9. What is the correct CSS syntax for making all the `<p>` elements bold?
(A) `p {font-weight:bold}` (B) `<p style="text-size:bold">`
(C) `p {font-style:bold}` (D) `<p style="font-size:bold">`
- (C) 10. Which CSS property is used to change the left margin of an element?
(A) left-margin (B) margin (C) margin-left (D) indent
- (D) 11. How to insert a comment in an XML document?
(A) `/* a comment */` (B) `// a comment` (C) `// a comment //` (D) `<!-- a comment -->`
- (B) 12. Which section is not interpreted by the XML parser?
(A) `<? ?>` (B) `<![CDATA[. . .]]>` (C) `<![PCDATA[. . .]]>` (D) none of above
- (D) 13. Which is not a correct name for an XML element?
(A) `<TITLE>` (B) `<Chapter>` (C) `<Section.1>` (D) none of above
- (B) 14. When an XML file is modified, what must be modified to display the changes?
(A) modify the HTML file (B) modify the CSS file (C) modify the XML file (D) modify the DTD file
- (A) 15. Which array is used to pass the parameters of a function in Perl?
(A) `@_` (B) `@$` (C) `@#` (D) `@%`
- (B) 16. If the address of an array is stored in \$value in Perl, how to get the values of this array?
(A) `&$value` (B) `@$value` (C) `\$value` (D) `$$value`

- (D) 17. How is a pound sign (#) encoded in a CGI query string? Its decimal ASCII value is 35.
 (A) @35 (B) #35 (C) #23 (D) %23
- (A) 18. How does a Perl program includes the CGI Perl module and use standard symbols?
 (A) use CGI q\$:standard\$; (B) use CGI "(:standard)"; (C) use CGI :standard; (D) None of above
- (A) 19. Which parameter is used to specify the lifetime of a cookie object in Perl?
 (A) expires (B) setLifeTime (C) setExpires (D) none of above
- (C) 20. Which directive in the httpd.conf file is used to customize the error responses?
 (A) ErrorResponse (B) ErrorReply (C) ErrorDocument (D) ErrorDoc
- (D) 21. In Java, the statement for connecting to the keyboard is
 (A) Stream in = System.in; (B) Stream System.in = new Stream(System.in);
 (C) Stream in = Stream(System.in); (D) Stream in = new Stream(System.in);
- (C) 22. Which following declaration is not correct in Java?
 (A) double duty; (B) double duty; (C) byte value = 192 (D) int start = 34, end = 99;
- (C) 23. What method of an Exception object returns a message string?
 (A) printError() (B) traceMessage() (C) getMessage() (D) printMessage()
- (B) 24. How is a column in a table which must have a value specified in SQL?
 (A) not empty (B) not null (C) default value (D) have value
- (A) 25. When the **delete** command is used in a SQL statement, which method of the **Statement** class should be invoked in JDBC?
 (A) executeUpdate() (B) executeQuery() (C) DeleteUpdate() (D) executeDelete()

Part 2: (125 points)

1. Briefly explain these terminologies. If they are acronyms, also write what they stand for. (24 points)
 - (a) **MIME** The Multi-purpose Internet Mail Extensions is an extension used to exchange different kinds of data files on the Internet such as audio, video, images, and application programs.
 - (b) **W3C** World Wide Web Consortium (W3C) is a consortium which specifies standards for the Web.
 - (c) **XHTML** The Extensible Hypertext Markup Language is a markup language to display documents on the Web as an application of the Extensible Markup Language (XML).
 - (d) **SGML** Standard Generalized Markup Language is a standard for how to specify a document markup language or tag set.
 - (e) **DTD** A document type definition (DTD) defines the valid data content syntax of XML elements.
 - (f) **CGI** The Common Gateway Interface (CGI) is the interface between a browser and software on the server.
 - (g) **Servlet Container** A servlet container is a program that manages servlets.
 - (h) **JSP** Java Server Page (JSP) is a technology that supports dynamic document through the use of servlets,

2. Describe the five most commonly used HTTP request methods. (10 points)

Ans:

GET	Return the contents of the specified document
HEAD	Return the header information for the specified document
POST	Execute the specified document, using the enclosed data
PUT	Replace the specified document with the enclosed data
DELETE	Delete the specified document

3. Describe three levels of style sheets and their applying range. (6 points)

Ans:

- (a) Inline style sheets apply to the content of a single tag.
- (b) Document-level style sheets apply to the whole body of a document.
- (c) External style sheets can apply to the bodies of any number of documents.

4. Describe two methods to get the command line arguments in Perl. Give an example for each method. (4 points)

Ans:

- (a) use shift. (\$a, \$b) = (shift, shift);
- (b) use @ARGV. (\$a, \$b) = @ARGV;

5. Consider the following Perl code. Write the results after running this code. (6 points)

```
@list = (8, 2, 7, 6, 9);
unshift @list, (4, 10);
print "$list[2] \n";
print scalar(@list), "\n";
print pop @list, "\n";
push @list, (5, 3);
print shift @list, "\n";
print "$#list \n";
```

8 7 9 4 6

6. Complete the following JDBC servlet with following functions:

- (a) Collect the information of a music CD from three text widgets with the attribute name = "Artist", name = "Title", and name = "Released_Year" respectively.
- (b) Insert that music CD into the Album table in a MySQL database with specifications as shown in the following two tables:

Host name	kira
Account name	music
Password	music123
Database	music

Field	Type
Artist	varchar(25)
Title	varchar(20)
Released_Year	integer

- (c) Display a list of the music CDs released in the same year as the one inserted above on the Web browser.

(15 points)

```

import javax.servlet.*;
import javax.servlet.http.*;
import java.io.*;
import java.sql.*;

public class Music extends HttpServlet {
    public void doPost(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
        try {
            Class.forName("com.mysql.jdbc.Driver").newInstance();
        } catch (Exception e) {}

        try {
            Connection conn = DriverManager.getConnection(
                "jdbc:mysql://kira/music?user=music&password=music123");

            response.setContentType("text/html");
            PrintWriter out = response.getWriter();

            String artist = request.getParameter("Artist");
            String title = request.getParameter("Title");
            String released_year = request.getParameter("Released_Year");

            out.println("<html><head><title>Music CD</title></head><body>");

            Statement stmt = conn.createStatement();
            String sql_cmd = "insert into Album values ('" +
                artist + "', '" + title + "', " + released_year + ")";

            int rows = stmt.executeUpdate(sql_cmd);

            sql_cmd = "select * from Album where Released_Year = " + released_year;

            ResultSet RS = stmt.executeQuery(sql_cmd);

            out.println("<table border>");
            out.println("<tr><tr><th>Artist</th><th>Title</th><th>Released Year</th></tr>");

            while (RS.next()) {
                out.println("<tr><td>" + RS.getString(1) + "</td>");
                out.println("<td>" + RS.getString(2) + "</td>");
                out.println("<td>" + RS.getFloat(3) + "</td></tr>");
            }
            out.println("</table>");
            out.println("</body></html>");
            out.close();

            RS.close();
            stmt.close();
            conn.close();
        } catch (SQLException E) {}
    }
}

```

7. Describe the three life-cycle methods called by the servlet engine. (6 points)

Ans:

init initializes a servlet and prepares it to respond to client requests.

service controls how the servlet responds to client requests.

destroy takes the servlet out of service, making it unavailable for client request.

8. Describe any five of the XML syntax rules. (10 points)

Ans:

- (a) All XML documents must have a root tag.
 - (b) All XML elements must have a closing tag.
 - (c) All XML elements must be properly nested.
 - (d) XML tags are case sensitive.
 - (e) Attribute values must always be quoted.
 - (f) Element names must follow naming rules.
9. (a) Write a HTML document with a text input form to collect a student's name and the score in an exam. This HTML document will request a CGI program `record.pl` in the `cgi` directory of your account at `kirk.cs.wichita.edu:6550`. (6 points)
- (b) Write the CGI program `record.pl` that writes the student's name and score into a file `record.dat` with the following format and then returns a message showing the student's name and scores to the Web browser. (6 points)

```
John Wayne, 92
Jayne Leno, 87
. . . . .
```

Hint: Use the following functions in CGI.pm: `param()`, `header()`, `start_html()`, and `end_html()`.

Ans:

```
(a) <html>
<head>
<title>HTML to call the CGI-Perl program record.pl</title>
</head>
<body>
<form action = "http://kirk.cs.wichita.edu:6550/cgi/cs655/record.pl"
      method = "POST">
<input type = "text" name = "name" size = "20">
<input type = "text" name = "score" size = "5">
<input type = "submit" value = "Submit">
</form>
</body>
</html>
```

```
(b) #!/usr/bin/perl -w
# record.pl - a CGI program that stores the name into record.dat and
#           returns the student's name and scores to the Web browser.

use CGI qw(:standard);

open (OUT, ">>record.dat");
my ($name, $score) = (param("name"), param("score"));
print OUT "$name, $score\n";
print header();
print start_html("Score Entry");
print "$name, $score\n";
print end_html();
```

- (c) Write a Perl program that reads the file `record.dat`, calculates the total number of attendance, the average of the scores, and the number in each 10 score range, and then prints out the result. Suppose the total score for the exam is 100. For example, a possible output would look like:
(12 points)

```
Attendance = 34
Average = 85.2058823529412
Score Range  Number
90 - 100      9
80 - 89       19
70 - 79       4
```

- (c) `#!/usr/bin/perl`

```
my @grading = ('0 - 59', '60 - 69', '70 - 79', '80 - 89', '90 - 100');
my ($sum, $attendo) = (0, 0);
while (<>) {
    my ($name, $score) = split(/,/);
    $attendo++;
    $sum += $score;
    $grades[($score - 50)/10]++;
}
$grades[4] += $grades[5];
print "Attendance = $attendo\n";
print "Average = ", $sum/$attendo, "\n";
print "Score Range  Number\n";
for ($i = $#grades; $i >= 0; $i--) {
    if ($grades[$i] > 0) {
        print "$grading[$i] \t $grades[$i]\n";
    }
}
}
```

10. Based on the last digital of your social security number, choose the object from the following table to specify: (20 points)

Last digital of your SSN	Objects
0, 1	TV Program
2, 3	Newspaper Article
4, 5	Company
6, 7	Speech
8, 9	Baseball Game

For example, if the last digital of your social security number is 5, then `Company` is the object you need to specify. Your object should have at least three attributes.

- Use SQL to create a database table to specify this object.
- Create an XML document to specify this object.
- Create a DTD document for this XML.
- Create an CSS document for this XML.

You can use the following information to create XML, DTD, and CSS.

```
<?xml version="1.0"?>
<!DOCTYPE note SYSTEM "InternalNote.dtd">
<!DOCTYPE note [
  <!ELEMENT payment (#PCDATA)>
  <!ATTLIST payment type CDATA "check">
]>
ad {display: block; color: blue;}
year, make, model {color: red; font-size: 16pt;}
```

Ans:

- (a) create table speech (
- ```
 title varchar(20),
 speaker varchar(40),
 place varchar(40),
 date date
);
```
- (b) <?xml version="1.0"?>
- ```
<speeches>
  <speech>
    <title>Jay Leno's Police Blotter</title>
    <speaker>Jayne Leno</performer>
    <place>Kansas Studium</performer>
    <date>2004-2-20</date>
  </speech>
</speeches>
```
- (c) <!DOCTYPE speeches [
- ```
<!ELEMENT speeches (speech)>
<!ELEMENT speech (title, speaker, place, date)>
<!ELEMENT title (#PCDATA)>
<!ELEMENT speaker (#PCDATA)>
<!ELEMENT place (#PCDATA)>
<!ELEMENT date (#PCDATA)>
]>
```
- (d) speeches
- ```
{
  background-color: gray;
  width: 100%;
}
speech
{
  display: block;
  margin-bottom: 30pt;
  margin-left: 0;
}
title, speaker, place, date
{
  color: yellow;
  font-size: 20pt;
}
```